Taxonomy Induction Using LLMs: An Enhanced Framework by Integrating Doubly-Checked Mechanism and Self-Evaluation Strategy

Abstract. Taxonomies, structured as tree hierarchies, are valuable for applications such as web retrieval, question-answering, and recommender systems. As existing taxonomy curation based on deep learning or pre-trained models rely heavily on a large amount of labeled data, which is extremely time-consuming and labor-intensive, the emergence of large language models (LLMs, *e.g.,* ChatGPT, GPT-4.0, LLaMA2) has made automatic taxonomy construction from texts highly desirable. However, relying solely on LLMs and prompt engineering makes it challenging to extract precise and comprehensive *is-a* relationships from texts. In response to this limitation, this paper aims to explore a well-designed approach to guide LLMs for better construction of taxonomies from texts. On one hand, we propose a doubly-checked mechanism to improve the quality of candidate nodes generated from texts. On the other hand, we utilize a modularized Chain-of-Thought prompting technique to break down hypernym identification into several sub-problems and employ the beam search-based self-evaluation strategy to enhance reliability. Specifically, self-evaluation constraint factors are introduced to score the reliability of reasoning chains generated through beam search, whereby selecting the most reliable chain as the final judgment. Experiments on datasets from various domains demonstrate the effectiveness and efficiency of our framework.

Keywords: Doubly-Checked Mechanism · Self-Evaluation Strategy · Taxonomy Induction · Large Language Models.

1 Introduction

A taxonomy is a hierarchical directed acyclic graph that organizes concepts or entities by "hypernym-hyponym" or *is-a* relations (*e.g.,* "apple" *is-a* "fruit"). This structure is crucial for downstream tasks such as text comprehension [17], personalized recommendations [2], and question answering [15]. To induce a taxonomy from texts, traditional methods mainly rely on manual construction or crowdsourced top-down manner, which is time-consuming, labor-intensive and difficult to scale. With the advancement of deep learning, extensive studies have focused on the automated taxonomy induction using pattern-based methods [7], word embeddings [11], and pre-trained language models (PLMs) [6]. Unfortunately, these approaches suffer from the issue of low coverage and semantic accuracy, impacting the performance of downstream tasks [4].

Recently, Large Language Models (LLMs) have shown their extraordinary ability to understand and generate texts in natural language, making them feasible solutions to address the issues of low coverage and semantic accuracy in taxonomy induction from texts. Consequently, existing studies have made some efforts to build taxonomies by leveraging the internal knowledge of LLMs, *e.g.,* identifying hierarchical relationships between given concepts while adhering to structural constraints [3] or employing an ensemble-based ranking filter in In-context learning [18]. However, these methods still make an insufficient exploration of the context information for each node, causing them to fail in processing special entities reasonably (*e.g.,* long-tail distributed entities and abbreviations) and ensuring the reliability of LLMs' response. These issues inevitably lead to poor accuracy in the taxonomy induction, impacting the performance of downstream applications.

To resolve these issues, this paper proposes a novel LLMs-based taxonomy induction framework, composed of two crucial steps, *i.e.,* the candidate nodes generation and the hypernym-hyponym relation detection. During the first step, there are typically special entities in the text to be extracted, requiring a high level of proficiency in contextual comprehension. Thus, we leverage the In-context learning technique combined with a doubly-checked mechanism to generate candidate terms from texts, thereby addressing issues related to these types of entities. Specifically, In-context learning provides a few examples of expected outputs for LLM, aiming to improve the performance of generating candidate nodes at a lower cost. Furthermore, to generate candidate nodes with greater accuracy, the doubly-checked mechanism is implemented through a multiple-choice prompt.

After obtaining the candidate terms, the second step aims to infer the hypernym-hyponym relationship between the two terms with the crucial operation being also understanding the contextual information. A beam search-based decoding strategy is employed to improve the stability and reasonableness of LLMs' predictions. Besides, a self-consistent evaluation method is further used to attain a state of reliability systematically. We regard the prediction probability of the model's self-evaluation at each step as the constraint factor for scoring each reasoning chain generated by beam search.

With the synergy of these two strategies, the proposed LLMs-based framework could dramatically improve the taxonomy induction task. To verify its effectiveness, we conduct extensive experiments on WordNet sub-taxonomies [1] and two large-scale, real-world taxonomies [13]. Experimental results demonstrate that the proposed method gains a large margin (about 12.67% on WordNet dataset) performance improvement compared with the state-of-the-art baseline. To summarize, the contributions of this paper could be listed as follows:

- We propose a doubly-checked mechanism along with In-context learning to address the special-entity issue in candidate term generation.
- We introduce the beam search decoding combined with a self-consistent evaluation strategy into the hyponymy detection to obtain a more reliable reasoning chain and a better identification result.

- Extensive experiments on three datasets from various domains verify that the proposed method significantly improves the performance of taxonomy induction from texts.

2 Related Work

Taxonomy induction typically involves identifying hypernyms and organizing these relationships into a hierarchical structure. Initially, traditional methods such as pattern-based techniques [10, 14, 16] and embedding-based approaches [8] were widely used. Additionally, Mao et al. [9] utilized reinforcement learning to integrate the phases of hypernym identification and hypernym organization. DNG [19] analyzes the inheritance and supplementary between node features in taxonomies and refines the taxonomy construction on the non-Gaussian space. Several works treat this task as a graph optimization problem, solved using the maximum spanning tree algorithm [1]. Shang et al. [12] applied a graph neural network, demonstrating improvements in large-scale taxonomy induction.

More recent approaches leverage the capabilities of large language models (LLMs) for taxonomy induction. Chen et al. [4] constructed taxonomic trees by predicting parenthood relations and optimizing these predictions into a maximum-spanning tree using pre-trained language models. Jain et al. [6] utilized treating taxonomy induction as sequence classification and sequence scoring tasks. Zeng et al. [18] constructed taxonomies by iteratively selecting relevant candidate entities for each layer and reducing errors through the Ensemble-based Ranking Filter. Langlais and Guo [5] proposed an automatic taxonomy evaluation metric based on pre-trained models. TaxonomyGPT [3] conducted taxonomy induction by leveraging the in-context learning capabilities of LLMs. The proposed method in this paper significantly improves structural accuracy by prompting large language models in the Chain-of-Thought style.

3 Taxonomy Induction based on LLMs

3.1 Overall framework

In this section, we elaborate on the proposed framework designed to address the taxonomy induction task. As illustrated in Fig. 1, the task is divided into two main steps. First, candidate terms are generated through a doubly-checked mechanism from texts using LLMs, as described in Section 3.2 Next, in Section 3.3, relation detection is performed via LLMs using a specially designed beam search and self-evaluation strategy. Once the nodes and relations are obtained, a taxonomy related to the texts can be constructed using the maximum spanning tree (MST) algorithm [1].

3.2 Doubly-Checked Mechanism for Candidate Node Generation

To enhance the generation of candidate terminology, we utilize in-context learning-based prompts. By providing a small number of examples, we can better guide the LLM in generating relevant terms in new scenario.

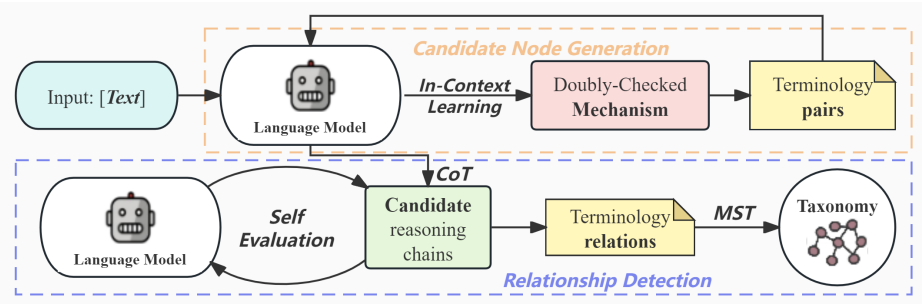

Fig. 1. Overall framework of the proposed method.

Typically, the issue of special entities is intractable for language models. For example, LLMs sometimes generate "single precision floating point format" in mistake instead of the ground truth "floating point". This may be attributed to their failure to grasp the crucial information of input texts. Besides, the meanings of some low-frequent terms are not directly observable, such as "h450". As these challenges demand great ability in contextual comprehension, a mechanism that facilitates fully utilizing textual information is an effective solution.

Inspired by Kozareva et al. [7], we design a doubly-checked mechanism for candidate node generation via In-context learning prompting to refine results gradually. As shown in Fig. 2, for a given text, an initial candidate term t_1 is first generated by the LLM using In-context learning prompts. To address the issues mentioned above, the candidate term t_1 undergoes a doubly-checked mechanism through a multiple-choice prompt:

1. If "Incorrect", the term is regenerated.
2. If "Correct", the detailed reference of the term is generated from the text.
3. If "Better Answer", a new term is generated and compared, allowing models to cross the threshold of multiple-choice questions and freely generate texts.

The final term is obtained after this verification mechanism.

3.3 Self-Evaluation Strategy for Relation Detection

To enhance the reliability and stability of reasoning chains generated by LLMs, we employ beam search decoding combined with a self-evaluation strategy to detect relationships between candidate term nodes. This approach allows us to generate multiple reasoning chains and score them accordingly.

CoT-based Relation Detection Relation detection via LLM adopts a Modularized-style CoT prompting in this paper, decomposing the hierarchical relationship detection task into a series of sub-problems. Given the input $(term_1, I(term_1); term_2, I(term_2))$, in which each of the terms is paired with its respective information $I(\cdot)$ in context, the LLM is asked to judge hierarchical relationships between two terms. However, we find that reasoning about their relationships

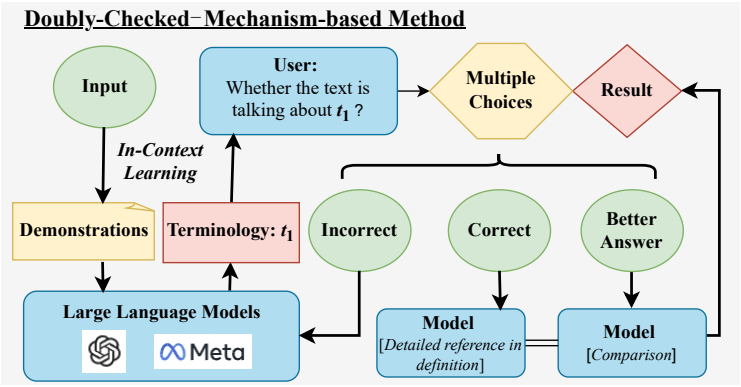

Fig. 2. Doubly-checked mechanism-based method for candidate node generation via In-context learning prompting.

requires more direct and concise informational text. Otherwise, the LLM's judgment will be affected by redundant information. Therefore, based on experience, this problem is decomposed into three sub-problems where CoT is used by guiding a step-by-step logical process:

1. Conclude the information of $term_1$ from the text;
2. Conclude the information of $term_2$ from the text;
3. Judge the hierarchical relationship between $term_1$ and $term_2$ based on the summarized information.

The Chain-of-Thought (CoT) prompting technique has been proven to be highly effective in reasoning tasks with LLMs. However, previous works mostly use a single reasoning sample generated by LLMs to derive results, which causes the generated reasoning chains unreliable and can even lead to incorrect answers if one of the steps is mistaken. To enhance the model's prediction accuracy, this paper also introduces beam search decoding.

Assuming the output examples for the first and second steps are s_{sum}, and s_{judge} for the third step, the outputs of the model for each step are r^1, r^2, r^3, the prompts are denoted as follows: $p^i = Prompt(term_i, I(term_i), s_{sum})(i = 1, 2), p^3 = Prompt(r^1, r^2, s_{judge})$.

In the specific task studied in this paper, the summarized information of $term_1$ and $term_2$ does not have a cause-and-effect relationship logically, but the summarized information of $term_1$ may influence the output of the summarized information of $term_2$. Therefore, we swap the order of p^1 and p^2 and perform the reasoning again, applying beam search to both the original and reversed sequences. After self-evaluation, the system ultimately generates $2k$ paths. Fig. 3 shows a variant of the beam search method ($k = 2$) designed for the specifics of this task, where k is an important parameter in beam search to determine the number of candidate sequences retained at each search step. Given that the information generated by LLMs may be unreliable, we consider a constraint factor $C(\cdot) \in [0, 1]$ to obtain higher quality and more reliable reasoning chains.

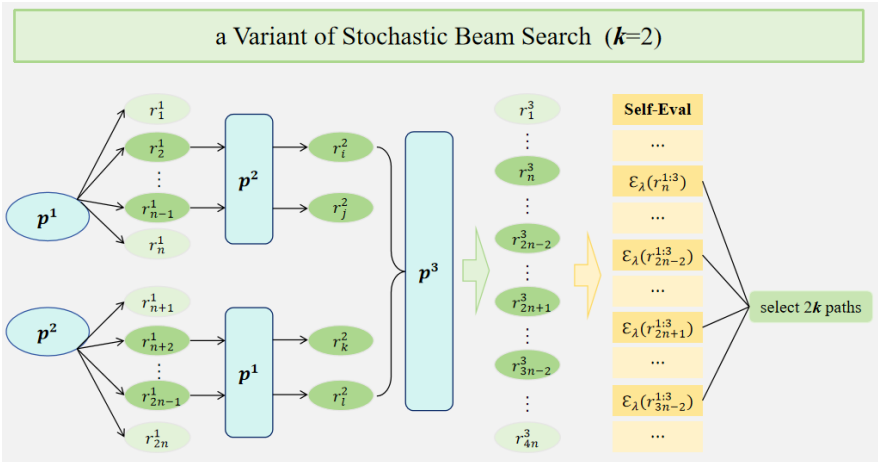

Fig. 3. Proposed variant of Stochastic Beam Search.

The subsequent sections provide an in-depth analysis of the multi-step reasoning process via beam search, as well as a detailed exposition of the constraint factors.

Multi-step Reasoning via Stochastic Beam Search When multiple-step reasoning with LLM, a T-step reasoning chain, donated as R, is composed of a series of output tokens. In this study, we set $T = 3$ and $R = [r^1, r^2, r^3] = r^{1:3}$. Formally, we can factorize the generation reasoning process $P(R = r^{1:T}|p^{1:T})$ in an auto-regressive manner:

$$P(R = r^{1:T}|p^{1:T}) = \prod_t P(r^t|p^t, r^{1:t-1}) \tag{1}$$

The decomposition of the reasoning process allows it to be viewed as a step-by-step decoding problem. One of the most serious issues in LLM-based reasoning is the potential unreliability and inaccuracy of each reasoning step generated by the model. Furthermore, this potential error can accumulate iteratively as the reasoning progresses. To mitigate the impact of this issue, this paper considers a self-evaluation-based constraint function at each reasoning step. This function, based on prompts and previous input-output texts of the current step, represents the reliability of the output at that step. Therefore, for the hierarchical relationship detection task, we can derive a decoding strategy that combines the probabilities from the large language model with the reliability probabilities, resulting in a new objective function $\Sigma_\lambda(R = r^{1:T})$:

$$\Sigma_\lambda(R = r^{1:T}) = \prod_t P_{LM}^\lambda(r^t|p^t, r^{1:t-1}) C^{1-\lambda}(r^t|p^t, r^{1:t-1}) \tag{2}$$

where P_{LM} is the language model distribution, $\lambda \in [0, 1]$ is a weight parameter used to balance the model prediction score and the faithfulness score. To obtain a high-quality inference chain with a higher score of $\Sigma_\lambda(R = r^{1:T})$, it

is natural to leverage greedy or stochastic beam decoding to approximate the reasoning chain with the maximum $\Sigma_\lambda(R = r^{1:T})$

Constraint Factor as Self-Evaluation We use LLMs to judge the faithfulness of r^t based on $r^{1:t-1}$. Specifically, the evaluation and generation models use the same back-end LLM with different prompts. We design a multiple-choice questioning for refinement of model predictions, denoted as $Prompt_C$, where the token-level probability of the option "*rational*" is adopted to calculate the faithfulness score:

$$C(R = r^t) = P_{LM}(A|Prompt_C, p^{1:t}, r^{1:t}) \tag{3}$$

4 Experiments

We evaluate the proposed taxonomy induction framework on four real-world benchmarks. To outline the experiments conducted in our paper, we raise three primary research questions that require resolution:

RQ1: How does the proposed method perform in taxonomy induction compared to state-of-the-art baselines?
RQ2: How does the doubly-checked mechanism affect the performance of the proposed method in taxonomy induction?
RQ3: What significant impact does the beam search-based self-evaluation strategy have on the effectiveness of the taxonomy induction task?

4.1 Experimental Settings

Datasets. We conducted our experiments on WordNet sub-taxonomy [1], DBLP and SemEval-Sci [13]. Specifically, WordNet consists of 761 non-overlapping taxonomies, each containing entities ranging from 11 to 50. DBLP is constructed from 156,000 paper abstracts within the field of computer science, with 176 concepts and 175 edges arranged in a 4-depth taxonomy. SemEval2016-Sci, presented as an 8-depth taxonomy with 429 concepts and 451 edges, is derived from a shared task of taxonomy induction in SemEval-2016.

Baseline Methods. We compare our proposed framework with the following supervised fine-tuning baseline methods:

- Graph2Taxo [12] leverages cross-domain graph structures and adopts constraint-based Directed Acyclic Graph (DAG) learning for taxonomy induction.
- CTP [4] fine-tunes the RoBERTa model to predict the probability of parent-child pair and integrates it into a graph using a maximum spanning tree algorithm for precise taxonomy induction.

Additionally, we also adopt the following unsupervised and in-context learning baseline methods for a comprehensive comparison:

Table 1. Performance on taxonomy induction on Ancestor-metrics.

Model	WordNet			DBLP			SemEval-Sci		
	P_a	R_a	$F1_a$	P_a	R_a	$F1_a$	P_a	R_a	$F1_a$
Graph2Taxo	<u>79.20</u>	47.80	59.60	47.85	30.23	37.05	82.45	36.15	50.27
CTP	69.30	**66.20**	<u>66.70</u>	45.62	41.39	43.40	52.41	33.88	41.16
RestrictMLM	23.23	25.69	24.09	-	-	-	63.33	<u>47.85</u>	<u>54.44</u>
LMScorer	37.50	47.64	41.59	17.14	21.54	19.04	48.80	33.24	39.51
TaxonomyGPT	62.97	41.77	48.95	28.98	14.40	17.15	53.09	31.84	39.07
Ours(LLaMA-2)	62.17	58.22	60.13	<u>76.50</u>	<u>71.44</u>	<u>73.88</u>	70.79	43.91	54.20
Ours(GPT-3.5)	**86.91**	<u>63.60</u>	**73.45**	**84.51**	**84.61**	**84.56**	<u>78.26</u>	**70.89**	**74.39**

- RestrictMLM [6] employs a "fill-in-the-blank" approach based on cloze statements to extract *is-a* relational knowledge from BERT.
- LMScore [6] treats taxonomy induction as a sentence scoring task using GPT-2, assessing the natural fluency of sentences.
- TaxonomyGPT [3] treats the taxonomy induction as a conditional text generation challenge. It represents the output taxonomy as a collection of sentences, each describing a parent-child relation within the output taxonomy.

For our proposed framework, we conduct experiments with GPT-3.5-turbo and LLaMA-2-7b-chat.

Evaluation Metrics. To evaluate the performance of all compared models, we adopt six evaluation metrics, *i.e.,* Ancestor-Precision, Ancestor-Recall, Ancestor-F1, Edge-Precision, Edge-Recall and Edge-F1 donated as P_a, R_a, $F1_a$, P_e, R_e and $F1_e$ respectively.

Ancestor-metrics compare the ancestor-descendant relations in the predicted taxonomy with those in the ground truth taxonomy. Specifically, if "Anc" donates the set of term pairs that have an "$is-ancestor$" relationship between them, we have:

$$P_a = \frac{|Anc_{pred} \cap Anc_{gold}|}{|Anc_{pred}|}, R_a = \frac{|Anc_{pred} \cap Anc_{gold}|}{|Anc_{gold}|}, F1_a = \frac{2P_a \cdot R_a}{P_a + R_a}. \quad (4)$$

Edge-metrics are more stringent compared to Ancestor-metrics. They evaluate the exactness of the predicted taxonomy by directly comparing the predicted edges with the gold standard edges. Similarly, if "E" represents the edge set of a taxonomy, we have:

$$P_e = \frac{|E_{pred} \cap E_{gold}|}{|E_{pred}|}, R_e = \frac{|E_{pred} \cap E_{gold}|}{|E_{gold}|}, F1_e = \frac{2P_e \cdot R_e}{P_e + R_e}. \quad (5)$$

4.2 Main Results (RQ1)

In our experiments, we compare the performance of our proposed method with five baseline methods on WordNet, DBLP and SemEval-Sci. From the experimental results shown in Table 1 and Table 2, we have three major observations.

Table 2. Performance on taxonomy induction on Edge-metrics.

Model	WordNet			DBLP			SemEval-Sci		
	P_e	R_e	$F1_e$	P_e	R_e	$F1_e$	P_e	R_e	$F1_e$
Graph2Taxo	**75.60**	37.00	49.70	46.63	28.49	35.37	**79.37**	34.52	46.87
CTP	53.30	49.80	51.50	38.21	33.73	35.83	31.18	29.42	30.27
RestrictMLM	24.17	25.65	24.89	-	-	-	45.79	46.19	45.99
LMScorer	36.27	38.48	37.34	25.84	26.12	25.98	42.20	42.58	42.39
TaxonomyGPT	49.20	43.85	46.24	34.27	22.17	25.97	39.59	36.84	38.01
Ours(LLaMA-2)	59.63	**60.34**	59.98	66.13	55.02	60.07	53.20	40.71	46.12
Ours(GPT-3.5)	73.14	54.36	**62.37**	**68.13**	**67.55**	**67.84**	65.90	**54.16**	**59.46**

First, the proposed method in this paper based on GPT-3.5 and Llama-2 outperforms the baselines on most evaluation metrics across the three datasets. They notably outperformed LMScorer by a significant margin. This observation underscores the great potential of leveraging LLMs endowed with exceptional text comprehension and generation capabilities for taxonomy induction.

Second, we find that even inducing taxonomy via prompting powerful LLMs, TaxonomyGPT(GPT-3.5) shows worse performance than methods such as CTP, which relies on fine-tuning BERT models, across all six metrics. This finding indicates that LLMs are sensitive to the way the prompt is designed for the specific task.

Finally, Graph2Taxo demonstrates quite high precision among the methods, showcasing its strength in utilizing lexical patterns as direct input features. However, its relatively lower recall reveals a significant trade-off, suggesting that although it excels in precision, it may not fully capture the complete range of taxonomic relations.

4.3 Effects of Doubly-Checked Mechanism (RQ2)

Taking GPT-3.5-turbo as an example, we conduct an ablation study to further verify the effectiveness of the doubly-checked mechanism. Compared with the results of the doubly-checked-free method displayed in Table 3, we find a steady improvement of all the evaluation metrics on three datasets by introducing the doubly-checked mechanism. Especially, when the doubly-checked mechanism was removed, the model's performance on the Edge-Recall metric significantly declined, even falling below that of some baselines. This finding indicates that the doubly-checked mechanism ensures a high-quality and high-coverage terminology generation for the following steps in the proposed framework.

4.4 Effects of Self-Evaluation Strategy (RQ3)

Additionally, we conduct an ablation study on the three benchmarks to probe "how does Self-Evaluation Strategy affects the performance of the proposed

Table 3. Effects of Doubly-checked Mechanism on taxonomy induction.

Datasets	Doubly-checked	Edge			Ancestor		
		P_e	R_e	$F1_e$	P_a	R_a	$F1_a$
WordNet	✗	68.53	37.10	40.14	81.24	57.88	67.60
	✓	73.14	54.36	62.37	86.91	63.60	73.45
DBLP	✗	67.14	52.93	59.19	81.88	75.35	78.48
	✓	68.13	67.55	67.84	84.51	84.61	84.56
SemEval-Sci	✗	62.12	23.77	34.38	72.51	70.49	71.49
	✓	65.90	54.16	59.46	78.26	70.89	74.39

method in taxonomy induction". Taking gpt-3.5-turbo as an example, Table 4 demonstrates the statistics of the ablation experiment, where a Self-Evaluation-free method is set to use zero-shot prompting during the relationship detection phase. As shown in the table, the application of the Self-Evaluation strategy improves $F1_e$ and $F1_a$ of the task on all three datasets. Specifically, "Precision" benefits a lot from the more stringent strategy, where results are derived from a carefully selected reasoning chain through self-evaluation.

Table 4. Effects of Self-Evaluation Strategy on taxonomy induction.

Datasets	Self-Evaluation	Edge			Ancestor		
		P_e	R_e	$F1_e$	P_a	R_a	$F1_a$
WordNet	✗	58.33	50.47	51.12	78.64	51.93	62.55
	✓	73.14	54.36	62.37	86.91	63.60	73.45
DBLP	✗	66.12	58.56	62.11	65.68	72.40	68.88
	✓	68.13	67.55	67.84	84.51	84.61	84.56
SemEval-Sci	✗	67.70	61.25	64.31	70.01	76.19	72.97
	✓	65.90	54.16	59.46	78.26	70.89	74.39

4.5 Case Study

This section utilizes a small-scale sample taxonomy from SemEval-Sci to briefly demonstrate the main experiment and the ablation experiments. Fig. 4 displays four taxonomies: the first is the ground truth taxonomy, and the remaining three are generated by the doubly-checked-free method, the self-evaluation-free method, and the proposed method in this paper, respectively. Blue edges or nodes in the ground truth taxonomy may be omitted in the output taxonomies, while red ones indicate redundancies compared to the ground truth.

In the small-scale sample, the proposed method performs well, with only one extra edge in the output taxonomy compared to the ground truth. In the

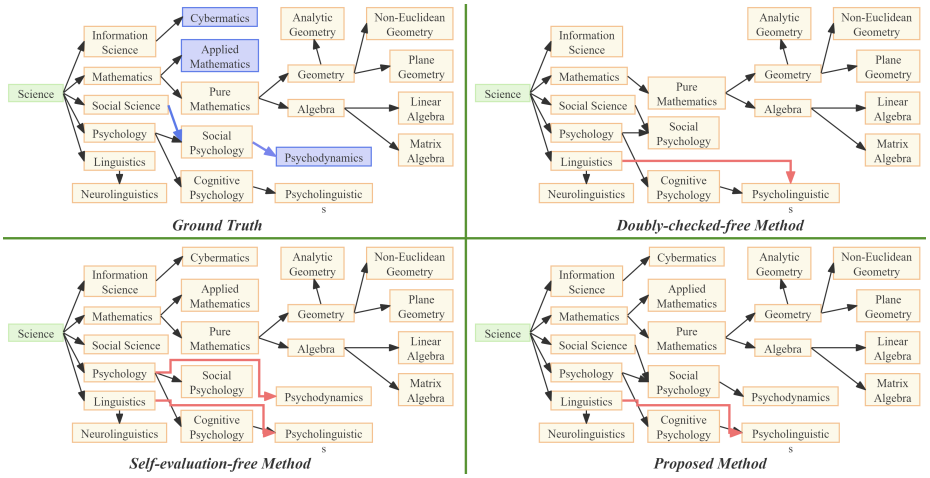

Fig. 4. Ground truth and output taxonomies on a sample of the SemEval-Sci dataset.

ablation experiments, the taxonomy induced by the proposed method covers more terminology nodes and correctly detects more relations.

5 Conclusion and Future Work

In this paper, we proposed an enhanced framework for taxonomy induction using large language models. By integrating a doubly-checked mechanism and a self-evaluation strategy, our method could effectively capture the context information during the taxonomy induction process. The experimental results indicated that the superiority of our approach in improving the structural accuracy. Additionally, ablation experiments demonstrated the effectiveness and potential of the two innovations in the taxonomy induction task.

In the future, we will focus on refining the mechanism and strategies and exploring their applicability to more domains and larger datasets. Additionally, considering the integration of other advanced techniques, such as structural information-based methods, could improve the efficiency of the taxonomy induction task using our proposed method.

References

1. Bansal, M., Burkett, D., de Melo, G., Klein, D.: Structured learning for taxonomy induction with belief propagation. In: ACL. pp. 1041–1051. ACL (2014)
2. Cao, Y., Wang, X., He, X., Hu, Z., Chua, T.S.: Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In: WWW. pp. 151–161 (2019)
3. Chen, B., Yi, F., Varró, D.: Prompting or fine-tuning? A comparative study of large language models for taxonomy construction. In: MODELS. pp. 588–596. IEEE (2023)

4. Chen, C., Lin, K., Klein, D.: Constructing taxonomies from pretrained language models. In: Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tür, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., Zhou, Y. (eds.) NAACL-HLT. pp. 4687–4700. ACL (2021)
5. Gao, T., Langlais, P.: Rate: a reproducible automatic taxonomy evaluation by filling the gap. CoRR **abs/2307.09706** (2023)
6. Jain, D., Anke, L.E.: Distilling hypernymy relations from language models: On the effectiveness of zero-shot taxonomy induction. In: Nastase, V., Pavlick, E., Pilehvar, M.T., Camacho-Collados, J., Raganato, A. (eds.) *SEM. pp. 151–156. ACL (2022)
7. Kozareva, Z., Riloff, E., Hovy, E.H.: Semantic class learning from the web with hyponym pattern linkage graphs. In: McKeown, K.R., Moore, J.D., Teufel, S., Allan, J., Furui, S. (eds.) ACL. pp. 1048–1056. ACL (2008)
8. Luu, A.T., Tay, Y., Hui, S.C., Ng, S.K.: Learning term embeddings for taxonomic relation identification using dynamic weighting neural network. In: Su, J., Duh, K., Carreras, X. (eds.) EMNLP. pp. 403–413. ACL (2016)
9. Mao, Y., Ren, X., Shen, J., Gu, X., Han, J.: End-to-end reinforcement learning for automatic taxonomy induction (2018)
10. Nakashole, N., Weikum, G., Suchanek, F.M.: PATTY: A taxonomy of relational patterns with semantic types. In: Tsujii, J., Henderson, J., Pasca, M. (eds.) EMNLP-CoNLL. pp. 1135–1145. ACL (2012)
11. Ristoski, P., Faralli, S., Ponzetto, S.P., Paulheim, H.: Large-scale taxonomy induction using entity and word embeddings. CoRR **abs/2105.01305** (2021)
12. Shang, C., Dash, S., Chowdhury, M.F.M., Mihindukulasooriya, N., Gliozzo, A.: Taxonomy construction of unseen domains via graph-based cross-domain knowledge transfer. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J.R. (eds.) ACL. pp. 2198–2208. ACL (2020)
13. Shen, J., Wu, Z., Lei, D., Shang, J., Ren, X., Han, J.: Setexpan: Corpus-based set expansion via context feature selection and rank ensemble. CoRR **abs/1910.08192** (2019)
14. Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. In: NIPS. pp. 1297–1304 (2004)
15. Wang, Y., Lipka, N., Rossi, R.A., Siu, A., Zhang, R., Derr, T.: Knowledge graph prompting for multi-document question answering. AAAI **38**(17), 19206–19214 (2024)
16. Wu, W., Li, H., Wang, H., Zhu, K.Q.: Probase: a probabilistic taxonomy for text understanding. In: Candan, K.S., Chen, Y., Snodgrass, R.T., Gravano, L., Fuxman, A. (eds.) SIGMOD. pp. 481–492. ACM (2012)
17. Yu, D., Zhu, C., Yang, Y., Zeng, M.: Jaket: Joint pre-training of knowledge graph and language understanding. AAAI **36**(10), 11630–11638 (2022)
18. Zeng, Q., Bai, Y., Tan, Z., Feng, S., Liang, Z., Zhang, Z., Jiang, M.: Chain-of-layer: Iteratively prompting large language models for taxonomy induction from limited examples. CoRR **abs/2402.07386** (2024)
19. Zhai, S., Wang, W., Li, Y., Meng, Y.: DNG: taxonomy expansion by exploring the intrinsic directed structure on non-gaussian space. In: AAAI. pp. 6593–6601. AAAI Press (2023)